Poster: Creating a User-Specific Perspective View for Mobile Mixed Reality Systems on Smartphones

Yuki Matsuda* Fumihisa Shibata Asako Kimura Hideyuki Tamura
Graduate School of Information Science and Engineering, Ritsumeikan University

ABSTRACT

We propose a method for creating a user-specific perspective view for mobile mixed reality (MR) systems that provides three-dimensional perception based on pseudo motion parallax. In general, most common mainstream interface for a hand-held AR/MR system is video-see-through style with a device-perspective view. On the other hand, our method provides a user-perspective view, which displays MR images according to the user's viewpoint and the position of the display. Namely, our method trims a scene behind the display and superimposes CG images in consideration of the user's point of view. This enables the user to perceive a stereoscopic effect with a non-3D display based on pseudo motion parallax. We implemented a prototype and tested our proposed method on smartphones.

Keywords: Mixed reality, user-specific perspective view, smartphone, motion parallax, head tracking.

Index Terms: H.5.1 [Information Interface and Presentation (e.g.,HCI)]: Multimedia Information Systems— Artificial, augmented, and virtual realities.

1 INTRODUCTION

In recent years, hand-held based AR/MR applications have become the mainstream in the field of AR/MR. However, most of them display AR/MR scene from the viewpoint of the camera on the hand-held device [1]. Of course, an ideal hand-held based AR/MR system displays a scene from the user's viewpoint, as if the hand-held device disappears and CG images are superimposed with perspective correctness (Figure 1). Geometrically correct user-perspective rendering for AR/MR systems could become a powerful depth cue for understanding three-dimensional structure of AR/MR world because it provides motion parallax. Hill et al. proposed a method for creating the augmented scene from the user's point of view [2]. However, they implemented the system with a single workstation equipped with GPU and a tablet display. Baričević et al. have also created a proof-of-concept prototype using a Kinect sensor, a Wiimote, and a workstation [1]. In this poster, we tried to implement a prototype on smartphones without any additional sensors.

2 CREATING A USER-SPECIFIC PERSPECTIVE VIEW

2.1 Overview

To create a user-specific perspective view, there are three challenges to be solved:

(a) How to estimate the user's head (eye) position
(b) How to estimate the six-degrees-of-freedom (6-DOF) display pose (position and orientation) in the world
(c) How to create the model of the real world

In this research, we tackle challenge (a) and (b) to realize a user-specific perspective view on smartphones, and we assume that the depth of the real world is always constant. In other words, we make an assumption that the background is a planar surface.

Above all, challenge (c) requires high-quality reconstruction of geometry. This kind of 3D reconstruction using active sensors, and cameras are well-studied in the field of computer graphics and computer vision. Methods using cameras need a lot of computational power for reconstructing geometry. One example of methods using active sensors is KinectFusion which was proposed by Izadi et al. [3]. However, this method requires additional sensors. Unfortunately, current smartphones does not equipped with active sensors, such as that of Kinect's, and do not have sufficient computational power. For these reasons, we did not tackle challenge (c).

2.2 Processing flow

In this section, we describe the processing flow of the proposed method that includes five steps.

Capturing Front Image The system captures the user's image (front image) using the front camera.

Estimating Position of User's Eyes The three-dimensional position of the user's eyes is estimated using the front image. Since the FOV of the front camera (*a*[degree]) and the width of the captured image (f_w[pixel]) are known and the pupillary distance of the user (*pd*[mm])can be measured in advance, the distance between the centers of the pupils in the captured image (*l*[pixel]) can be estimated by image processing. From these values, the distance between the front camera and the user's eye (*z'*[mm]) can be calculated by the following equation (Figure 2 - 3).

$$z' = \frac{f_w \cdot pd}{2l\tan\frac{a}{2}} \tag{1}$$

Next, the distances *x*'[mm] and *y*'[mm] from the optical axis of the front camera are calculated by equation (2) and (3) respectively where *x*[pixel] and *y*[pixel] are the position of the user's eyes in the captured image(Figure 3).

$$x' = \frac{l \cdot x}{pd} \tag{2}$$

$$y' = \frac{l \cdot y}{pd} \tag{3}$$

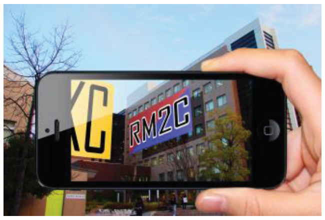


Figure 1: The smartphone seems like a picture frame if a fully user- perspective view is achieved.

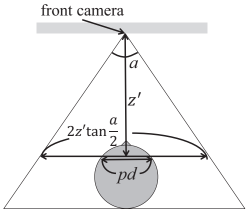


Figure 2: This figure shows the setting for estimating the position of the user's eyes.

---

* email: y_matuda@rm.is.ritsumei.ac.jp

IEEE Symposium on 3D User Interfaces 2013
16-17 March, Orlando, FL, USA
978-1-4673-6098-2/13/$31.00 ©2013 IEEE

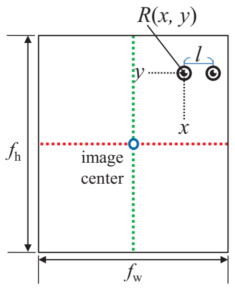


Figure 3: Front image setting. $R(x, y)$ is the position of the user's right eye.

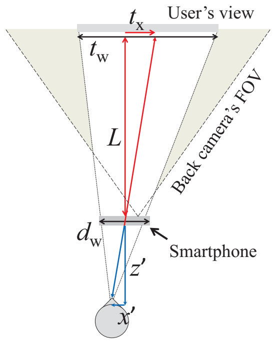


Figure 4: Horizontal direction. To calcuate t_w and t_x, a homologous triangle is used. Calculation of the vertical direction is done in a similar fashion.

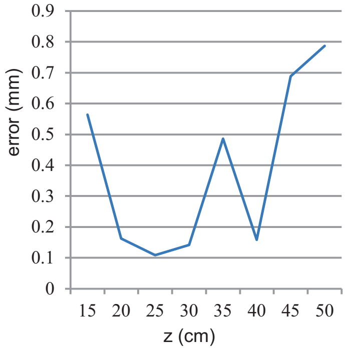


Figure 5: The errors on the z-axis were less than 0.8mm. They are almost ignorable because the average was about 0.4mm.

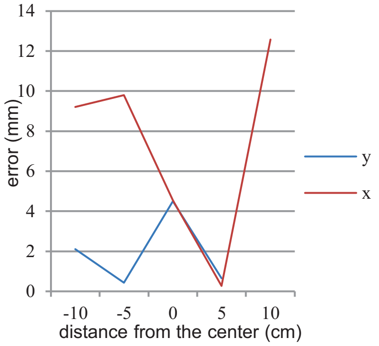


Figure 6: The origin is the center of the display. The errors on the x-axis and the y-axis are less than 13mm. The minimum values are not at the origin because the front camera is not mounted at the center of the display.

Capturing Background Image The system captures the background image using the back camera.

Estimating Position of Smartphone In this step, the system estimates the pose (position and orientation) of the smartphone based on the existing geometric registration method.

Rendering MR Image In the final step, the system trims the background image according to the position of the user's eyes. The width (t_w[pixel]) and height (t_h[pixel]) of the trimmed image are determined by the following equation (4) and (5), where d_w[mm] and d_h[mm] are the actual width and height of the smartphone's display respectively and L[mm] is the distance between the smartphone and the background surface, which we assume in advance (Figure 4).

$$t_w = \frac{d_w(z'+L)}{z'} \tag{4}$$

$$t_h = \frac{d_h(z'+L)}{z'} \tag{5}$$

Next, the center of the trimmed image (t_x, t_y) are calculated by the following equations.

$$t_x = \frac{x' \cdot L}{z'} \tag{6}$$

$$t_y = \frac{y' \cdot L}{z'} \tag{7}$$

Finally, the system superimposes CG images onto the trimmed background image.

3 EXPERIMENTS

3.1 Implementation

We implemented our proposed method on a pair of iPhone4S. The reason is because current iOS does not support to use the front and the back cameras simultaneously and switching the camera mode from the front view to the back view takes about 0.5s. One phone estimates the position of the user's eyes and sends it to the other phone via Bluetooth connection. The other phone estimates its pose and displays an MR image according to the user's eye.

3.2 Results

Our current implementation runs between 12 to 15 fps. Figure 5 and 6 show the errors between the estimated position of the user's eye and its ground truth. From this result, we think that there is no problem while the user holds the phone within arm's reach. Figure 7 shows the result when the user moves slightly from side to side or back and forth.

4 CONCLUSION

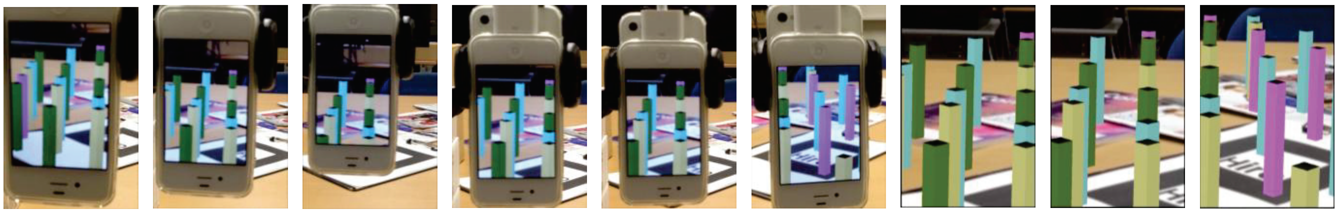
In this poster, we proposed a method for creating a user-specific perspective view for MR. Our method provides the user three-dimensional perception based on pseudo motion parallax. We implemented a prototype on a couple of iPhone4S because of the limitation of current iOS. Currently, we assume that the depth of the real world is constant. One idea for overcoming this limitation is to make use of smartphones equipped with dual cameras.

REFERENCES

[1] D. Baričević, *et. al.* A hand-held AR magic lens with user-perspective rendering. In *Proceedings of the 11th IEEE Int'l Symp. on Mixed and Augmented Reality (ISMAR 2012)*, pages 197 – 206, 2012.

[2] A. Hill, *et. al.* Virtual transparency: introducing parallax view into video see-through AR. In *Proceedings of the 10th IEEE Int'l Symp. on Mixed and Augmented Reality (ISMAR 2011)*, pages 239 – 240, 2011

[3] S. Izadi, *et. al.* KinectFusion: real-time 3D reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symp. on User Interface Software and Technology (UIST 2011)*, pages 559 – 568, 2011.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 7: The images taken near the user position show the user-specific perspective view. When the user moves front-back direction,the field of view of the smartphone changes naturally ((a) - (c)). When the user moves slightly from side to side, the user can perceive a stereoscopic effect based on motion parallax ((d) - (f) and their screenshots (g) - (i)).